The TOPGUN Approach to Software Acquisition Education

Mark E. Nissen
Naval Postgraduate School

The Naval Postgraduate School's TOPGUN approach augments the education of military officers with the leading principles, practices, tools, and techniques associated with software program management. The four-unit course involves student education, training, and research, in addition to direct experience through the TOPGUN software acquisition project. Because so many software acquisitions fail, this education and experience can be critical to surviving a program manager's first encounter with acquisition. As in the dogfight, surviving one's first software acquisition represents the key challenge to returning to learn from a second and third.

Software is not only ubiquitous, it is imperative. Information technology is deeply embedded within nearly every high-performance military system in use today, and many of the highest-performing weapons systems could not be mission effective without software. This applies to flight control systems, radar tracking and fire control systems, command, control, communications, computers, and intelligence systems, autonomous mission planning systems and many others. Even the infantryman is beginning to rely on software for enhanced battlefield performance. Indeed, the software share of weapons systems life-cycle cost has grown to be quite large, and it is growing at an increasing rate.

However, when compared with weapons systems hardware, software engineering represents a nascent discipline that reflects a high variance of methods, tools, technologies, and results. Coupled with steadily increasing computing power, software requirements perennially stretch the technological envelope and frequently are tied to schedules that challenge even the most mature engineering disciplines. Hence, software engineering represents one of the highest risk areas in modern weapons systems development, and its share of program risk increases exponentially with the growing dependence on weapons systems software. This makes embedded weapons systems software an extremely difficult acquisition problem.

The Naval Postgraduate School (NPS) has recently redesigned a graduate course from the acquisition curriculum to address this problem by augmenting the education of military officers with the leading principles, practices, tools, and techniques associated with software program management. The specific missions of this course are to help mitigate the difficulties inherent in software-intensive systems acquisition, to educate students in the principles and techniques of software acquisition and engineering, and to spark increased innovation in the acquisition process. The course, entitled "Embedded Software Acquisition," is offered to NPS graduate students—predominately military, but international officers and civilian professionals also attend—who are in residence at the Monterey, Calif. campus and enrolled in the program management (816) curriculum. Several elements of the course may also lend themselves to (synchronous) distance education.

The software acquisition course is comprised of four primary elements: education, training, research, and experience. Educationally, the course addresses the key principles of software acquisition and engineering, and takes a process-innovation approach. Students learn how and why software is unique and study textbook practice and practical cases associated with its engineering and innovation in the commercial sector and the Department of Defense (DoD). Training involves direct exposure to the DoD processes for software acquisition and assimilation of the best commercial practices and military lessons learned for procurement and program management. Students learn to critique—and perform in—the current acquisition system, with an emphasis on anticipating continuous acquisition reform and effecting process innovation. Through research papers, the students investigate the innovation aspects of software acquisition they are likely to encounter as program managers.

The experiential component involves "hands-on" usage of current tools and methods employed in software acquisition as students are given direct responsibility for managing and executing software development projects. These tools and methods include function and feature points, COnstructive COst MOdel estimating (COCOMO), Statistical Modeling and Estimation of Reliability Functions for Software reliability prediction, software capability assessment, metrics, and collaboration. Students learn to integrate and apply their relevant education and training through integrated product teams (IPTs) that are responsible for software planning, development, maintenance, and support.

TOPGUN Approach

The TOPGUN approach refers to the aerial combat tactics course that inspired a movie by the same name. Military research that focused on the disappointing aerial combat kill ratios of previous conflicts found that aviators who had experienced and survived even a small number of dogfighting engagements were exponentially more likely to survive their next one than were inexperienced pilots. Thus, the TOPGUN program was developed to provide a realistic training environment for students to *experience* and learn from aerial combat in a relatively safe environment.

Clearly, this approach is not limited to aerial combat tactics. The integration of principled education and realistic experience represents a well-accepted and effective pedagogical technique. This approach represents a centerpiece of the NPS software acquisition course, which as noted above, incorporates principles of software program management with the realistic experience of managing a software development project.

Because every graduate from the program management curriculum is increasingly likely to be responsible for some aspect of weapons systems software acquisition—even if assigned responsibility for a "hardware" program—this course helps to prepare them for their first encounter. In TOPGUN tradition, the students "train like they fight, and fight like they train." Moreover, because so many software acquisitions fail, this education and training can be critical to surviving the first encounter; as with the dogfight, we feel that surviving one's first software acquisition represents the key challenge to returning to learn from a second and third.

Course Requirements

Course requirements include readings, case analyses, an examination, a research paper, student synopses, and class participation in addition to the TOPGUN project. Each of the requirements for this four-unit course are summarized here; a more thorough description can be accessed through the course Web pages (http://web.nps.navy.mil/~menissen/mn3309).

Readings

Students are assigned more than a thousand pages of reading material that is required for adequate class preparation. They quickly learn that their (graded) class participation suffers if they fail to keep pace with the readings, and they are taught advanced volume-reading techniques for assistance with this requirement. Readings include the Software Technology Support Center *Guidelines for Successful Acquisition and Management of Software-Intensive Systems* as a text, numerous General Accounting Office cases that address software management, an abundance of on-line readings, and handouts from current journals and periodicals. Although the course is primarily directed toward education as opposed to training, the assigned material also includes the Defense Acquisition Workforce Improvement Act content from the software acquisition management series.

Case Analyses

As noted above, numerous software management cases are assigned as an integral part of the course readings and integrated into case-method lectures. One or two encompassing cases are thoroughly analyzed and presented by students working in small teams (three or four people). The purpose of the case analyses is to ground students' course knowledge in complex and relevant software programs. This requires the application of concepts, principles, tools, and techniques learned in class in addition to the analysis of real-world programs and design of program-management solutions.

Examination

A comprehensive examination tests students' ability to integrate the readings with lectures, cases, and discussions and to critically apply the principles, concepts, and applications covered in the course. The examination mostly tests students' higher-level knowledge and abilities (as opposed to narrow details) and requires understanding, application, synthesis, and some creativity and design (as opposed to short-term memorization and "data dumping"). Unlike case analyses and the course project, which involve teamwork, the examination measures only individual effort.

Research Paper

Each student must write a research paper (approximately 5,000 words) that investigates a relevant aspect of software acquisition or engineering innovation. This helps students specifically tailor their learning to a topic of direct or immediate relevance. It also provides the opportunity to work toward innovation, the results of which can be taken directly to the job and applied after graduation. Recent papers have investigated the use of requisite variety for weapons systems prioritization, agent-based contract management, reengineering the software acquisition process, and other personalized investigations. Because the course is taken midway through the master's program at NPS, students are encouraged to focus on topics that help them orient and focus their thesis research.

Student Synopses and Participation

Oral and written communications represent important managerial skills for military and business leaders, and with the advent of oral contracting, critical listening and interaction skills are now becoming increasingly important in acquisition. To practice and enhance these skills, students work in two-person teams to prepare written synopses of selected course material, then present this material to the class. Presenters are expected to effectively communicate the key elements of this material to classmates, who must in turn interact with the presenters to demonstrate a reasonable understanding and appreciation for the key points. Active class participation is strongly encouraged, and students strive to relate their relevant operational experience to the principles and practices covered in class.

TOPGUN Project

In addition to accredited graduate education and the requirements outlined above, the TOPGUN project probably best distinguishes this software acquisition course from others. The project entails team activities that involve software acquisition and software engineering. Half the teams serve as program management offices (PMOs) for Web-based software projects and the other teams perform as contractors to develop the required software systems. Although the software is technically simple and the project scope is small—as is appropriate for a quarter-long course—the students must go through most of the required acquisition and engineering steps, e.g., a mock Defense Acquisition Board and Milestone II review, Software Development Plan, Preliminary Design Review and Critical Design Review, Request for Proposal preparation and analysis, proposal preparation and source selection, in addition to managing the development of the software. Many outside software professionals have commented on the realism of the project environment, particularly in terms of the ambiguity, uncertainty, schedule pressure, metrics-tracking, and IPT-coordination difficulties that are encountered even on this technically simple project.

As an interesting twist (and pedagogical departure from reality?), the incrementally developed software is fielded at the project midpoint—midway through the course, ready or not! The teams then switch roles and maintain the software in a post-deployment software support (PDSS) mode; that is, those on the PMO teams become “contractors” and maintain the software, while the original contractor teams take over the PDSS activities, such as site operation, training, and enhancement. The students never fail to express their amazement of how this change of perspective affects their attitudes, and students on both sides of the contract quickly reflect back on their performance in the other role with greater empathy and understanding of how interrelated software acquisition and engineering truly are.

The project involves developing a Web site on software acquisition (http://web.nps.navy.mil/~com) that is not intended to be spectacular in any respect; rather, it provides a focus and some minimal technical challenge for the course project, and it requires the students to integrate, organize, and present their learning about software management to the world. Students take this requirement seriously—although they appear to enjoy the project tremendously—and produce insightful summaries of key principles, practices, tools, and techniques associated with software management. They also provide a valuable set of lessons learned for future classes as an approach to organizational learning.

Results

This course produces a number of results. At its conclusion, many students express feelings of confidence combined with humility, for example. They seem to appreciate the power and value of advance planning, close integration of software acquisition with engineering, teaming, metrics, and “inch pebble” (small milestone) management, and they feel much better prepared to take on a software management job after graduation. At the same time, they appreciate how difficult it can be to manage a technology-focused IPT and program, even for a technically simple product. Faculty colleagues comment on the students’ comfortable familiarity with software concepts in their other courses and their ability to extend some of the useful techniques to non-software program management problems.

In the end, nearly all students indicate that they are better prepared to manage software-intensive programs as a result of the course—and I concur. If they can “come back alive” from their first engagement with a *real* software program, all the work required to make this kind of course effective will pay off. With students from the first class graduating this year and being assigned to challenging weapons systems programs, we should soon see whether our TOPGUN approach to software acquisition fulfills our high expectations for it. As a professor dedicated to constant improvement, I welcome any feedback that can help further this goal. ◆

About the Author

Mark Nissen is a professor of information systems and acquisition management at the Naval Postgraduate School and leads the school’s program of acquisition research. He is the course coordinator for MN3309, Embedded Software Acquisition, and also teaches courses on decision support and information technology acquisition, in addition to advanced graduate seminars on process innovation, intelligent agents, and like topics. His research is directed toward the application of knowledge systems to innovate acquisition processes, with current work focused on intelligent acquisition agents. Before beginning his academic career, he acquired a dozen years management experience in the aerospace industry and was a supply officer in the Naval Reserve.

Naval Postgraduate School
555 Dyer Road, Code SM/NI
Monterey, CA 93943-5000
Voice: 408-656-3570
Fax: 408-656-3407
E-mail: Mnissen@nps.navy.mil